

290  
291  
  
292  
293  
  
294  
295  
  
296  
  
297  
  
298  
299  
  
300  
  
301  
302  
  
303  
  
304  
  
305  
306  
307  
308  
309  
310  
311  
312  
313

**[0021]**     What is claimed is:

1. A method for reserving a non-volatile cache for explicit control by an application comprising:  
reserving a first portion of the cache for application memory requests based at least in part on a predetermined set of functions that are supported by a driver for application calls; and  
reserving a second portion of the cache for application memory requests.
2. The method of claim 1 wherein the predetermined set of functions comprises: Allocate, Get, Set, and Free.
3. The method of claim 1 wherein the predetermined set of functions allow for direct or indirect application calls.
4. A method for reserving a non-volatile cache for explicit control by an application comprising:  
reserving a first portion of the cache for application memory requests based at least in part on a predetermined set of functions that are supported by a driver for application calls; and  
reserving a second portion of the cache to be used as a disk cache.
5. The method of claim 4 wherein the predetermined set of functions comprises: Allocate, Get,

314 Set, and Free.

315

316 6. The method of claim 1 wherein the predetermined set of functions allow for direct or indirect  
317 application calls.

318

319 7. An apparatus comprising:

320 a non-volatile cache, coupled to a main memory and a mass storage; and

321 the non-volatile cache to support a predetermined set of functions that are supported by a  
322 driver for application calls and a bit is set and cleared per affected cache-line in the cache-line  
323 metadata in the cache and the data allocation is done on a cache-line granularity.

324

325 8. The apparatus of claim 7 wherein the predetermined set of functions comprise: Allocate, Get,  
326 Set, and Free.

327

328 9. The apparatus of claim 7 wherein the predetermined functions allow for direct or indirect  
329 application calls.

330

331 10. The apparatus of claim 7 wherein the apparatus is to be implemented in either: a memory  
332 controller, a chipset, or an application specific integrated circuit (ASIC).

333 11. The apparatus of claim 8 wherein the non-volatile cache, in response to an Allocate function,  
334 will:

335 determine whether a predetermined number of bytes can be reserved,

336

337 if so, to identify cache-lines to use to reserve the predetermined number of bytes,

338  
339  
340  
341  
342  
343  
344  
345  
346  
  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
  
368  
  
369  
370  
371  
372  
373  
374

flush the cache-lines if they are dirty and mark them as empty,  
pin these cache-lines, and  
return a pointer to a structure that identifies the cache-lines reserved for this request.

12. The apparatus of claim 8 wherein the non-volatile cache, in response to a Set function, will:  
determine that input paramaters are valid (not null) and a data region referenced is in range,  
identify the cache-lines to use,  
copy data from a data Buffer to the applicable cache lines and mark these lines valid (not empty).

13. The apparatus of claim 8 wherein the apparatus is supervised by a driver in a software algorithm.

14. The apparatus of claim 8 wherein the non-volatile cache, in response to a Get function, will:  
determine that input parameters are valid (not null) and a date region referenced is in range,  
identify the cache-lines to use and determine if they are valid (not empty), and  
copy data from the applicable cache lines into a data Buffer.

15. The apparatus of claim 8 wherein the non-volatile cache, in response to a Free function, will:  
determine that input parameters are valid (not null),  
unpin the cache-lines,  
and Mark the cache lines as invalid

16. An apparatus comprising:  
a non-volatile cache, coupled to a main memory and a mass storage; and  
the non-volatile cache to support a predetermined set of functions that are supported by a  
driver for application calls and the cache is specifically utilized for an application and the non-  
volatile cache does not require pin bits.

375 17. The apparatus of claim 16 wherein the predetermined set of functions comprise: Allocate,  
376 Get, Set, and Free.

377  
378 18. The apparatus of claim 16 wherein the predetermined functions allow for direct or indirect  
379 application calls.

380  
381 19. The apparatus of claim 16 wherein the apparatus is to be implemented in either: a memory  
382 controller, a chipset, or an application specific integrated circuit (ASIC).

383 20. The apparatus of claim 17 wherein the cache, in response to the predetermined set of the  
384 functions, will:  
385 reserve a section of the cache for the application; and  
386 invoke a cache manager on a pre-reserved portion of the cache to support the predetermined set  
387 of functions.  
388

389 21. An article of manufacture comprising:  
390 a machine-readable medium having a plurality of machine readable instructions, wherein  
391 when the instructions are executed by a system, the instructions provide to manage a cache  
392 memory for:  
393 allocating a first portion of the cache memory for application memory requests based at  
394 least in part on a predetermined set of functions that are supported by a driver for application  
395 calls; and  
396 initializing at least one byte of the first portion of the cache memory in response to the  
397 predetermined set of functions;  
398 reading at least one byte of the first portion of the cache memory in response to the  
399 predetermined set of functions; and

400           deallocating at least one byte of the first portion of the cache memory in response to the  
401           predetermined set of functions.

402

22. The article of manufacture of claim 21 wherein the predetermined set of functions  
comprises: Allocate, Get, Set, and Free.

23. The article of manufacture of claim 21 wherein predetermined functions allow for  
direct or indirect application calls.